SFNet: Learning Object-aware Semantic Correspondence

Junghyup Lee, Dohyung Kim, Jean S Ponce, Bumsub Ham

SFNet: Learning Object-aware Semantic Correspondence

Junghyup Lee[1,*] Dohyung Kim[1,*] Jean Ponce[2,3] Bumsub Ham[1,†]
[1]Yonsei University [2]DI ENS [3]INRIA

Abstract

We address the problem of semantic correspondence, that is, establishing a dense flow field between images depicting different instances of the same object or scene category. We propose to use images annotated with binary foreground masks and subjected to synthetic geometric deformations to train a convolutional neural network (CNN) for this task. Using these masks as part of the supervisory signal offers a good compromise between semantic flow methods, where the amount of training data is limited by the cost of manually selecting point correspondences, and semantic alignment ones, where the regression of a single global geometric transformation between images may be sensitive to image-specific details such as background clutter. We propose a new CNN architecture, dubbed SFNet, which implements this idea. It leverages a new and differentiable version of the argmax function for end-to-end training, with a loss that combines mask and flow consistency with smoothness terms. Experimental results demonstrate the effectiveness of our approach, which significantly outperforms the state of the art on standard benchmarks.

1. Introduction

Establishing dense correspondences across images is one of the fundamental tasks in computer vision [5, 30, 36]. Early works have focussed on handling different views of the same scene (stereo matching [19, 36]) or adjacent frames (optical flow [4, 5]) in a video sequence. Semantic correspondence algorithms (*e.g.*, SIFT Flow [30]) go one step further, finding a dense flow field between images depicting different instances of the same object or scene category. This is very challenging especially in the presence of large changes in appearance/scene layout and background clutter. Classical approaches to semantic correspondence [3, 20, 26, 30, 47] typically use an objective function involving fidelity and regularization terms. The fidelity term encourages hand-crafted features (*e.g.*, SIFT [32], HOG [7], DAISY [45]) to be matched along a dense flow field between images, and the regularization term makes it smooth while aligning discontinuities to object boundaries. Although they have proven useful in various computer vision tasks including object recognition [10, 30], semantic segmentation [26], co-segmentation [44], image editing [8], and scene parsing [26, 50], hand-crafted features do not capture high-level semantics (*e.g.*, appearance and shape variations), and are not robust to image-specific details (*e.g.*, texture, background clutter, occlusion).

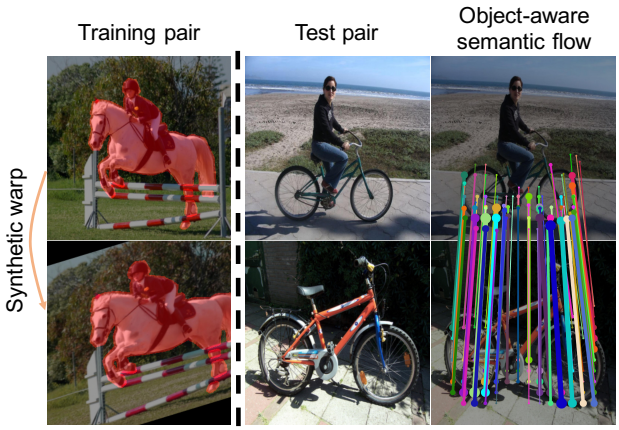


Figure 1: We use pairs of warped foreground masks obtained from a single image (left) as a supervisory signal to train our model. This allows us to establish object-aware semantic correspondences across images depicting different instances of the same object or scene category (right). No masks are required at test time. (Best viewed in color.)

Convolutional neural networks (CNNs) have allowed remarkable advances in semantic correspondence in the past few years. Recent methods using CNNs [6, 16, 23, 24, 27, 35, 40, 41, 42, 51] benefit from rich semantic features invariant to intra-class variations, achieving state-of-the-art results. Semantic flow approaches [6, 16, 27, 35, 51] attempt to find correspondences for individual pixels or patches. They are not seriously affected by non-rigid deformations, but are easily distracted by background clutter. They also require a large amount of data with ground-truth correspondences

*Equal contribution. †Corresponding author.
[1]School of Electrical and Electronic Engineering, Yonsei University, Seoul, Korea.
[2]Département d'Informatique de l'ENS, ENS, CNRS, PSL Research University, Paris, France.

for training. Although pixel-level semantic correspondences impose very strong constraints, manually annotating them is extremely labor-intensive and somewhat subjective, which limits the amount of training data available [14]. An alternative is to learn feature descriptor only [6, 27, 35] or to exploit 3D CAD models provided by rendering engines [51]. Semantic alignment methods [23, 24, 40, 41, 42] on the other hand formulate semantic correspondence as a geometric alignment problem and directly regress parameters of a global transformation model (*e.g.*, affine and thin plate spline) between images. This leverages self-supervised learning where ground-truth parameters are generated synthetically using random transformations with, however, a higher sensitivity to non-rigid deformations. Moreover, background clutter prevents focussing on individual objects and distracts estimating the transformation parameters. To overcome this problem, recent methods alleviate the influence of distractors by inlier counting [41] or an attention process [42].

In this paper, we present a new approach to establishing an object-aware semantic flow and propose to exploit binary foreground masks as a supervisory signal (Fig. 1). Our approach builds upon the insight that correspondences of high quality between images allow to segment common objects from background. To implement this idea, we introduce a new CNN architecture, dubbed SFNet, that outputs a semantic flow field at a sub-pixel level. We leverage a new and differentiable version of the argmax function, a kernel soft argmax, together with mask/flow consistency and smoothness terms to train SFNet end-to-end, establishing object-aware correspondences while filtering out distracting details. Our approach has the following advantages: First, it is a good compromise between current semantic flow and alignment methods, since masks are available for large dataset, and they give a good set of constraints. Exploiting binary foreground masks *explicitly* for training makes it possible to focus on learning correspondences between prominent objects and scene elements. Note that no masks are required at test time. Second, our method establishes a dense non-parametric flow field (*i.e.*, semantic flow), which is more robust to non-rigid deformations than a parametric regression (*i.e.*, semantic alignment). Finally, the kernel soft argmax enables training the whole network end-to-end, and hence our approach further benefits from high-level semantics specific to the task of semantic correspondence. The main contributions of this paper can be summarized as follows:

- We propose to exploit binary foreground masks directly, that are widely available and can be annotated more easily than the pixel-level ground truth, to learn semantic flow by incorporating them into loss functions.
- We introduce a kernel soft argmax, making it less susceptible to multi-modal distributions while providing a differentiable flow field at a sub-pixel level.
- We set a new state of the art on standard benchmarks for semantic correspondence, clearly demonstrating the effectiveness of our approach to exploiting foreground masks. We additionally provide an extensive experimental analysis with ablation studies.

To encourage comparison and future work, our code and models are available online: https://cvlab-yonsei.github.io/projects/SFNet.

2. Related work

Correspondence problems cover a broad range of topics in computer vision including stereo, motion analysis, object recognition and shape matching. Giving a comprehensive review on these topics is beyond the scope of this paper. We briefly review representative works related to ours.

Classical approaches have focussed on finding sparse correspondences, *e.g.*, for instance matching [32] or establishing dense matches between nearby views of the same scene/object, *e.g.*, for stereo matching [19, 36] and optical flow estimation [4, 5]. Unlike these, semantic correspondence methods estimate dense matches across pictures containing different instances of the same object or scene category. Early works on semantic correspondence focus on matching local features from hand-crafted descriptors, such as SIFT [3, 20, 26, 30], DAISY [47] and HOG [14, 44, 46], together with spatial regularization using graphical models [20, 26, 30, 44] or random sampling [1, 47]. However, designing hand-crafted features while considering high-level semantics is extremely hard, and computing similarities between them is easily distracted *e.g.*, by clutter, texture, occlusion and appearance variations. There are many attempts to estimate correspondences robust against background clutter or scale changes between objects/object parts, by using object proposals as candidate regions for matching [14, 46] or performing matching in scale space [38].

Recently, image features from CNNs have shown the powerful capacity of representing high-level semantics and the robustness to appearance and shape variations [17, 29, 43]. Long *et al*. [31] first apply CNNs to establish semantic correspondences between images. They follow the same procedure as the SIFT Flow [30] method, but exploit off-the-shelf CNN features trained for ImageNet classification tasks due to a lack of training datasets with pixel-level annotations. This problem can be alleviated by synthesizing ground-truth correspondences from 3D models [51] or augmenting the number of match pairs in a sparse keypoint dataset using interpolation [44]. More recently, the PF dataset [15] has been released providing 1300+ image pairs of 20 image categories with ground-truth annotations from the PASCAL 2011 keypoint dataset [2]. This enables learning local features [16, 27, 35] specific to the task of semantic correspondence. Although these approaches using CNN features outperform early methods by large margins, loss functions for training do not involve a spatial regularizer mainly due to a lack of differentiability of the flow field. In contrast,

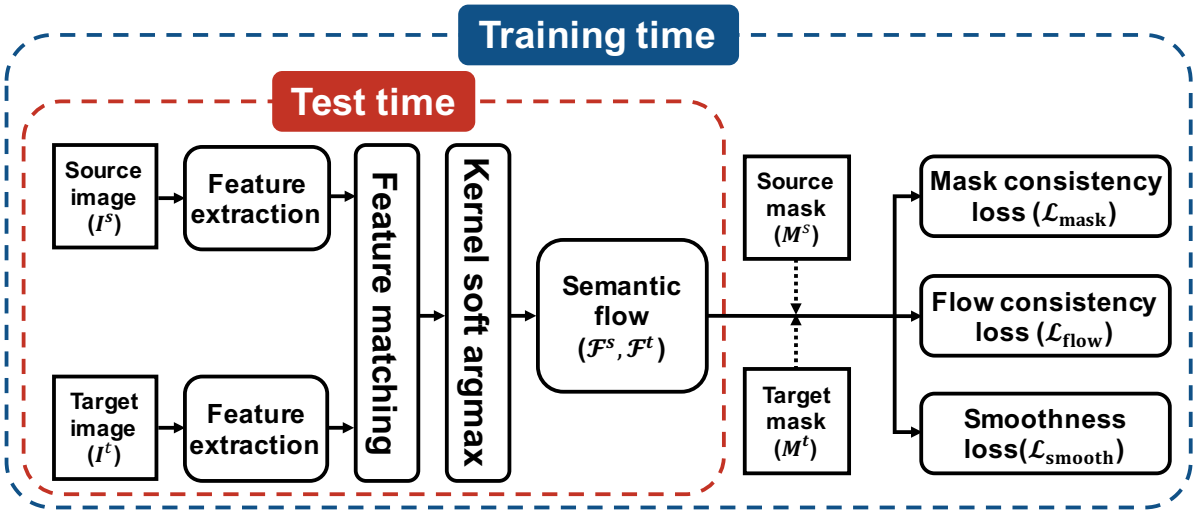


Figure 2: **Overview of SFNet.** SFNet inputs a pair of source and target images, I^s and I^t, and extracts local features using a siamese network. It then computes pairwise matching scores between features and establishes semantic flow, $\mathcal{F}^s$ and $\mathcal{F}^t$, for source and target images, respectively, by the kernel soft argmax. At training time, corresponding foreground masks, M^s and M^t, for source and target images, respectively, are used to compute mask consistency, flow consistency, and smoothness terms. See text for details.

our flow field is differentiable, allowing to train the whole network with a spatial regularizer end-to-end.

Several recent methods [23, 24, 40, 41, 42] formulate semantic correspondence as a geometric alignment problem using parametric models. In particular, these methods first compute feature correlations between images, and they are fed into a regression layer to estimate parameters of a global transformation model (*e.g.*, affine, homography, and thin plate spline) to align images. This makes it possible to leverage self-supervised learning [24, 40, 41, 42] using synthetically generated data and to train the entire CNNs end-to-end. These approaches apply the same transformation to all pixels, which has the effect of an implicit spatial regularization, providing smooth matches and often outperforming semantic flow methods [6, 14, 16, 27, 51]. However, they are easily distracted by background clutter and occlusion [24, 40], since correlations between pairs of features are noisy and include outliers (*e.g.*, between different backgrounds). Although this can be alleviated by using attention models [42] or suppressing outlier metches [41], global transformation models are highly sensitive to non-rigid deformations or local geometric variations. In this context, our method avoids this problem by establishing semantic correspondences directly from feature correlations.

Similar to ours, many methods [23, 27, 50, 51] leverage object bounding boxes or foreground masks to learn semantic correspondence. They, however, do not incorporate the object location prior explicitly into loss functions. They instead use the prior for pre-processing training samples, *e.g.*, generating positive/negative training pairs [23, 27] or limiting the candidate regions for matching [50, 51]. In contrast, we incorporate the prior directly into loss functions to train the network, outperforming the state of the art by a significant margin.

3. Approach

In this section, we describe our approach to establishing object-aware semantic correspondences including the network architecture (Sec. 3.1) and loss functions (Sec. 3.2). An overview of our method is shown in Fig. 2.

3.1. Network architecture

Our model is fully convolutional and mainly consists of three parts (Fig. 2): We first extract features from source and target images, I^s and I^t, using a siamese network where each sub-network has the same structure with shared parameters. We then compute matching scores between all pairs of local features in the two images, and assign the best match for each feature by the kernel soft argmax. All components are differentiable, allowing us to train the whole network end-to-end. In the following, we describe the network architecture for source to target matching in detail. A target to source matching is similarly computed.

Feature extraction and matching. We exploit a ResNet-101 [17] trained for ImageNet classification [9] for feature extraction. Although such CNN features give rich semantics, they typically fire on highly discriminative parts for classification. This may be less adequate for feature matching that requires capturing a spatial deformation for fine-grained localization. We thus use additional adaptation layers to extract features specific to the task of semantic correspondence, transforming them to be highly discriminative w.r.t both appearance and spatial context. This gives a feature map of size $h \times w \times d$ for each image that corresponds to

$h \times w$ grids of d-dimensional local features. We then apply L2 normalization to the individual d-dimensional features. As will be seen in our experiments, the adaptation layers boost the matching performance drastically.

Matching scores are computed using the dot product between local features, resulting in a 4-dimensional correlation map of size $h \times w \times h \times w$ as follows:

$$c(\mathbf{p}, \mathbf{q}) = f^s(\mathbf{p})^\top f^t(\mathbf{q}), \tag{1}$$

where we denote by $f^s(\mathbf{p})$ and $f^t(\mathbf{q})$ d-dimensional features at positions $\mathbf{p} = (p_x, p_y)$ and $\mathbf{q} = (q_x, q_y)$ in the source and target images, respectively.

Kernel soft argmax layer. We can assign the best matches by applying the argmax function over a 2-dimensional correlation map $c_\mathbf{p}(\mathbf{q}) = c(\mathbf{p}, \mathbf{q})$, w.r.t all features $f^t(\mathbf{q})$ at each spatial location $\mathbf{p}$. However, the argmax is discrete and not differentiable. The soft argmax [18, 25] computes an output by a weighted average of all spatial positions with corresponding matching probabilities. Although it is differentiable and enables fine-grained localization at a sub-pixel level, the output is influenced by all spatial positions, which is problematic especially in the case of multi-modal distributions.

We introduce a hybrid version, the *kernel soft argmax*, that takes advantage of both the soft and discrete argmax. Concretely, it computes correspondences $\phi(\mathbf{p})$ for individual locations $\mathbf{p}$ as an average of all coordinate pairs $\mathbf{q} = (q_x, q_y)$ weighted by a matching probability $m_\mathbf{p}(\mathbf{q})$ as follows.

$$\phi(\mathbf{p}) = \sum_\mathbf{q} m_\mathbf{p}(\mathbf{q})\mathbf{q}. \tag{2}$$

The matching probability $m_\mathbf{p}$ is computed by applying a spatial softmax function to a L2-normalized version $n_\mathbf{p}$ of the correlation map $c_\mathbf{p}$:

$$m_\mathbf{p}(\mathbf{q}) = \frac{\exp(\beta k_\mathbf{p}(\mathbf{q}) n_\mathbf{p}(\mathbf{q}))}{\sum_{\mathbf{q}' \in n_\mathbf{p}} \exp(\beta k_\mathbf{p}(\mathbf{q}') n_\mathbf{p}(\mathbf{q}'))}, \tag{3}$$

where $k_\mathbf{p}$ is a 2-dimensional Gaussian kernel centered on the position, computed by applying the discrete argmax to $n_\mathbf{p}$[1]. That is, we perform element-wise multiplication between the score map $n_\mathbf{p}$ and kernel $k_\mathbf{p}$, and then apply the softmax function. This retains the scores $n_\mathbf{p}$ near the output of the discrete argmax while suppressing others, having the effect of restricting the range of averaging in (2) and making it less susceptible to multi-modal distributions (*e.g.*, from ambiguous matches in background clutter and repetitive patterns) while maintaining differentiability. β is a "temperature" parameter adjusting a distribution of the softmax output. Note that as it becomes larger, the softmax function approaches the discrete one with one clear peak, but this may cause an unstable gradient flow at training time. Different from [18, 25], we perform L2 normalization on the 2-dimensional correlation map $c_\mathbf{p}$, adjusting the matching scores $f^s(\mathbf{p})^\top f^t(\mathbf{q})$ to a common scale before applying the softmax function. Note that the normalization is particularly important for semantic alignment methods [23, 40, 41, 42] (see, for example, Table 2 in [40]) but for different reasons. It penalizes features having multiple highly-correlated matches, boosting the scores of discriminative matches.

[1] At training time, we compute the kernel $k_\mathbf{p}$ every iterations and no gradients are propagated through the discrete argmax, making the matching probability $m_\mathbf{p}$ differentiable.

3.2. Loss

We exploit binary foreground masks as a supervisory signal to train the network, which gives a strong object prior. To this end, we define three losses that guide the network to learn object-aware correspondences without pixel-level ground truth as

$$\mathcal{L} = \lambda_{\text{mask}}\mathcal{L}_{\text{mask}} + \lambda_{\text{flow}}\mathcal{L}_{\text{flow}} + \lambda_{\text{smooth}}\mathcal{L}_{\text{smooth}}, \tag{4}$$

which consists of mask consistency $\mathcal{L}_{\text{mask}}$, flow consistency $\mathcal{L}_{\text{flow}}$ and smoothness $\mathcal{L}_{\text{smooth}}$ terms, balanced by the weight parameters (λ_{mask}, λ_{flow}, λ_{smooth}). In the following, we describe each term in detail.

Mask consistency loss. We define a flow field $\mathcal{F}^s$ from source to target images as

$$\mathcal{F}^s(\mathbf{p}) = \phi(\mathbf{p}) - \mathbf{p}. \tag{5}$$

Similarly, a flow field $\mathcal{F}^t$ from target to source images are defined as $\phi(\mathbf{q}) - \mathbf{q}$. We denote by M^s and M^t binary masks of source and target images, respectively. The values of 0 and 1 in the masks indicate background and foreground regions, respectively. We assume that reconstructing foreground/background masks by feature matching requires computing reliable similarities between features and dense correspondences of a high quality. To implement this idea, we transfer the target mask M^t by warping [22] using the flow field $\mathcal{F}^s$ and obtain an estimate of the source mask $\hat{M}^s$ as follows.

$$\hat{M}^s = \mathcal{W}(M^t; \mathcal{F}^s). \tag{6}$$

Here, we denote by $\mathcal{W}$ a warping operator using a flow field, *e.g.*, $\mathcal{W}(M^t; \mathcal{F}^s)(\mathbf{p}) = M^t(\mathbf{p} + \mathcal{F}^s(\mathbf{p}))$. We then compute the difference between the source mask M^s and its estimate $\hat{M}^s$. Similarly, we reconstruct the target mask $\hat{M}^t$ from M^s using the field $\mathcal{F}^t$ and compute its difference from M^t. Accordingly, we define the mask consistency loss as

$$\mathcal{L}_{\text{mask}} = \sum_{i \in \{s,t\}} \left(\frac{1}{|N^i|} \sum_\mathbf{p} (M^i(\mathbf{p}) - \hat{M}^i(\mathbf{p}))^2 \right), \tag{7}$$

where $|N^i|$ is the number of pixels in the mask M^i. Although the mask consistency loss does not enforce not aligning the background with anything, it prevents matches from foreground to background regions and vice versa by penalizing them. This encourages correspondences to be established between features within foreground masks and background

masks, guiding our model to learn object-aware correspondences. Note that the mask consistency loss does not restrict a many-to-one matching. That is, it does not penalize a case when many foreground features in an image are matched to a single one in other image, since binary masks do not give a positional certainty of correspondences.

Flow consistency loss. A flow consistency loss measures consistency between flow fields $\mathcal{F}^s$ and $\mathcal{F}^t$ within foreground masks defined as

$$\mathcal{L}_{\text{flow}} = \sum_{i\in\{s,t\}} \left(\frac{1}{|N_F^i|} \sum_{\mathbf{p}} ||(\mathcal{F}^i(\mathbf{p}) + \hat{\mathcal{F}}^i(\mathbf{p})) \odot M^i(\mathbf{p})||_2^2 \right), \tag{8}$$

where $|N_F^i|$ is the number of foreground pixels in the mask M^i, and

$$\hat{\mathcal{F}}^s = \mathcal{W}(\mathcal{F}^t; \mathcal{F}^s), \tag{9}$$

which aligns the flow field $\mathcal{F}^t$ with respect to $\mathcal{F}^s$ by warping. $\hat{\mathcal{F}}^t$ is computed similar to (9). We denote by $\|\cdot\|_2$ and $\odot$ the L2 norm and element-wise multiplication, respectively. The multiplication is applied separately for each x and y component. The flow consistency term favors a one-to-one matching, spreading flow fields over foreground regions and alleviating the many-to-one matching problem in the mask consistency loss. For example, when the flow fields are consistent with each other, $\mathcal{F}^s$ and $\hat{\mathcal{F}}^s$ have the same magnitude with opposite directions. Similar ideas have been explored in stereo matching [13, 49] and optical flow [33, 52], but without considering appearance and shape variations. It is hard to incorporate this term in current semantic flow methods based on CNNs [6, 16, 27] mainly due to a lack of differentiability of the flow field. Recently, Zhou *et al*. [51] exploit cycle consistency between flow fields, but they regress correspondences directly from concatenated features from source and target images and do not consider background clutter. In contrast, our method establishes a differentiable flow field by computing feature similarities explicitly while considering background clutter.

Smoothness loss. The differentiable flow field also allows to exploit a smoothness loss, which has been widely used in classical energy-based approaches [20, 26, 30]. We define a smoothness loss using the first-order derivative of the flow fields $\mathcal{F}^s$ and $\mathcal{F}^t$ as

$$\mathcal{L}_{\text{smooth}} = \sum_{i\in\{s,t\}} \left(\frac{1}{|N_F^i|} \sum_{\mathbf{p}} ||\nabla \mathcal{F}^i(\mathbf{p}) \odot M^i(\mathbf{p})||_1 \right), \tag{10}$$

where $\|\cdot\|_1$ and ∇ are the L1 norm and the gradient operator, respectively. This regularizes (or smooths) flow fields within foreground regions while not accounting for correspondences at background.

4. Experiments

In this section we present a detailed analysis and evaluation of our approach including ablation studies on different losses and network architectures.

4.1. Implementation details

Following [41, 42], we use CNN features from ResNet-101 [17] trained for ImageNet classification [9]. Specifically, we use the networks cropped at `conv4-23` and `conv5-3` layers, respectively. This results in two feature maps of size $20 \times 20 \times 1024$ and $10 \times 10 \times 2048$, respectively, for a pair of input images of size 320×320, which gives a good compromise between localization accuracy and high-level semantics. Adaptation layers are trained with random initialization, separately for each feature map in a residual fashion [17]. To compute residuals, we add 5×5 and 3×3 convolutional layers with padding on top of `conv4-23` and `conv5-3`, respectively, with batch normalization [21] and the ReLU [29]. The residuals are then added to the corresponding input features. With the resulting two feature maps of size $20 \times 20 \times 1024$ and $20 \times 20 \times 2048$[2], we compute pairwise match scores and then combine them by element-wise multiplication, resulting in a correlation map of size $20 \times 20 \times 20 \times 20$. We do not finetune the whole network due to a lack of training data, and train adaptation layers only. We empirically set the temperature parameter β to 50 and standard deviation σ of Gaussian kernel $k_{\mathbf{p}}$ to 5. Other parameters for losses are fixed to all experiments ($\lambda_{\text{mask}} = 3$, $\lambda_{\text{flow}} = 16$, $\lambda_{\text{smooth}} = 0.5$). We use a grid search to set these parameters, and choose the ones that give the best performance on the validation split of the PF-PASCAL dataset [15, 41]. At test time, we upsample a flow field of size 20×20 using bilinear interpolation.

4.2. Training

Training our network requires pairs of foreground masks for source and target images depicting different instances of the same object category. Although the TSS [44] and Caltech-101 [12] datasets provide such pairs, the number of masks is not enough to train our network [44] or there is a lack of background clutter [12]. Our model trained with these datasets suffers from a overfitting problem or may not generalize well for other images containing clutter. Motivated by [24, 34, 40, 42], we generate pairs of source and target images synthetically from single images by applying random affine transformations and use the synthetically warped pairs as training samples. Corresponding foreground masks are also transformed with the same transformation parameters. Contrary to [24, 34, 40, 42], our model does not perform a parametric regression, and thus it does not require ground-truth transformation parameters for training. We use the Pascal VOC 2012 segmentation dataset [11] that consists of 1,464, 1,449, and 1,456 images for training, validation and test, respectively. We exclude 122 images from train/validation sets that overlap with the test split in the PF-PASCAL [15], and train our model with the corresponding 2,791 images. We augment the training dataset by

[2] We upsample the features adapted from `conv5-3` using bilinear interpolation.

Type		Methods	PCK ($\alpha = 0.1$) WILLOW	PASCAL
Hand-crafted	F	DeepFlow [39]	0.20	0.21
	F	GMK [10]	0.27	0.27
	F	SIFTFlow [30]	0.38	0.33
	F	DSP [26]	0.29	0.30
	F	HOG+PF-LOM [15]	0.56	0.45
CNN-based	A	(T) ResNet-101+CNNGeo [40]	0.68	0.68
	A	(T) ResNet-101+A2Net [42]	0.69	0.67
	A	(T+P) ResNet-101+WS-SA [41]	<u>0.71</u>	<u>0.72</u>
	F	(B+P) FCSS+PF-LOM [27]	0.58	0.46
	F	(M) ResNet-101+Ours	**0.74**	**0.79**

Table 1: Quantitative comparison with the state of the art on the PF-WILLOW [14] and the test split of the PF-PASCAL [15, 16] in terms of the average PCK. We measure the PCK scores with height and width of the bounding box size. All numbers except for the methods of [40, 41, 42] are taken from [15, 42]. Numbers in bold indicate the best performance and underscored ones are the second best. We denote by "F" and "A", respectively, semantic flow and semantic alignment methods. The characters in parentheses are types of a supervisory signal for training; T: Transformation parameters; P: Image pairs depicting different instances of the same object category; B: Bounding boxes; M: Foreground masks.

horizontal flipping and color jittering. Note that we do not use segmentation masks, provided by the Pascal VOC 2012 dataset, that specify the class of the object at each pixel. We instead generate binary foreground masks using all labeled objects, regardless of image categories and the number of object, at training time. We train our model with a batch size of 16 about 7k iterations, giving roughly 40 epochs over the training data. We use the Adam optimizer [28] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. A learning rate initially set to 3e-5 is divided by 5 after 30 epochs. All networks are trained end-to-end using `PyTorch`[37].

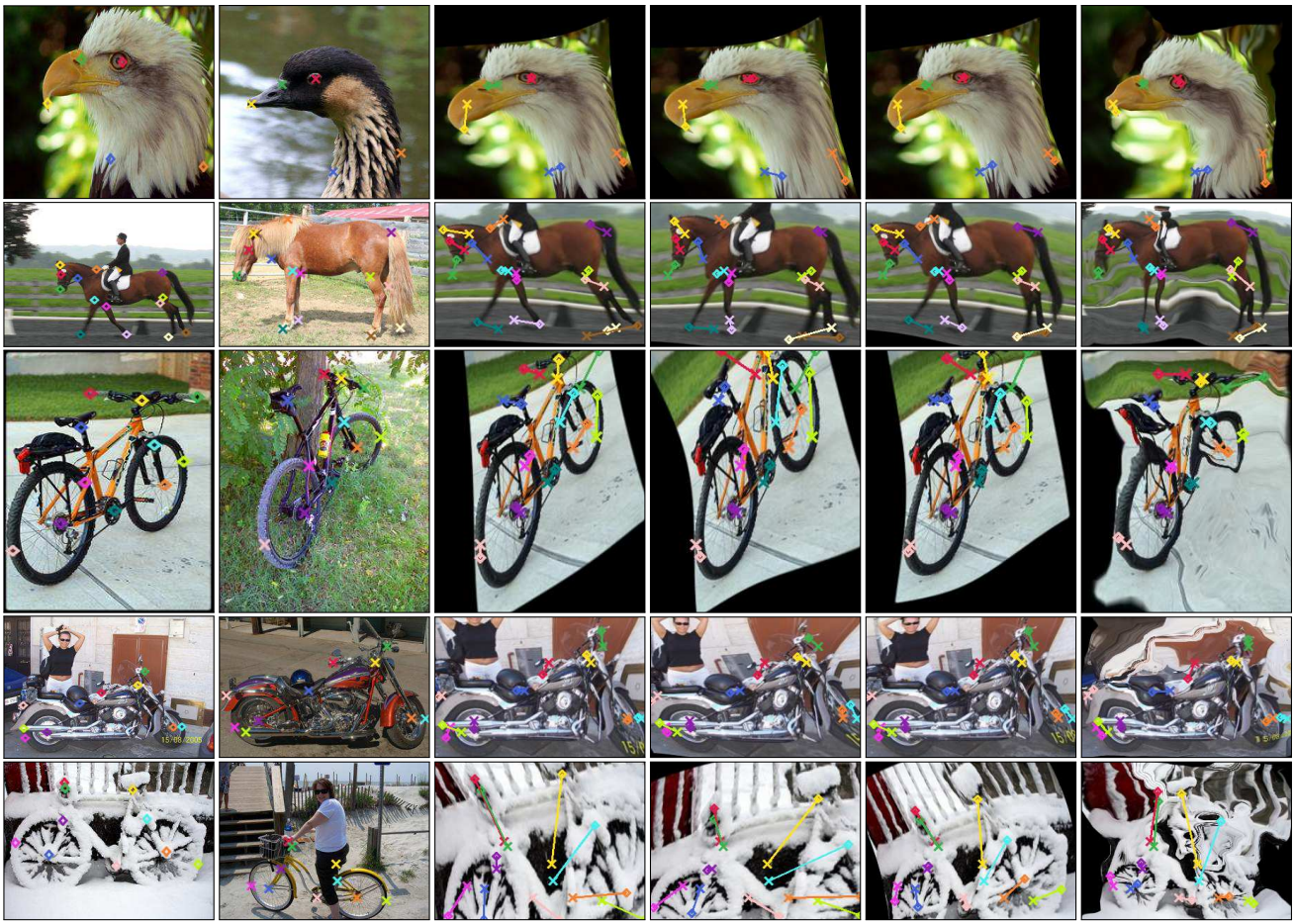
4.3. Results

We compare our model to the state of the art on semantic correspondence including hand-crafted and CNN-based methods with the following three benchmark datasets: PF-WILLOW [14], PF-PASCAL [15], and Caltech-101 [12]. The results for all comparisons have been obtained from the source code or models provided by the authors.

PF-WILLOW & PF-PASCAL. The PF-WILLOW [14] and PF-PASCAL [15] datasets provide 900 and 1,351 image pairs of 4 and 20 image categories, respectively, with corresponding ground-truth object bounding boxes and keypoint annotations. These benchmarks are more challenging than other datasets [12, 44] for semantic correspondence evaluation, featuring different instances of the same object class in the presence of large changes in appearance and scene layout, clutter and scale changes between objects. To evaluate our model, we use the PF-WILLOW and the test split of the PF-PASCAL provided by [16, 41] corresponding roughly 900 and 300 image pairs, respectively. We use the probability of correct keypoint (PCK) [48] to measure the precision of overall assignment, particularly at sparse keypoints of semantic relevance. We compute the Euclidean distances between warped keypoints using an estimated dense flow and ground truth, and count the number of keypoints whose distances lie within $\alpha \max(h, w)$ pixels, where $\alpha = 0.1$ and h and w are the height and width of the object bounding box, respectively.

We show in Table 1 the average PCK scores for the PF-WILLOW and PF-PASCAL datasets, and compare our method with the state of the art including hand-crafted [10, 15, 26, 30, 39] and CNN-based methods [27, 40, 41, 42]. The PCK scores in [40, 41, 42] are obtained by the provided models (affine + TPS). All other numbers are taken from [15, 42]. From this table, we observe four things: (1) Our model outperforms the state of the art by a significant margin in terms of the PCK especially for the PF-PASCAL datasets. In particular, it shows better performance than other object-aware methods [15, 27] that focus on establishing region correspondences between prominent objects. A plausible explanation is that establishing correspondences between object proposals is susceptible to shape deformations. (2) We can clearly see that our model gives better results than semantic alignment methods [40, 41, 42] on both datasets, but performance gain for the PF-PASCAL dataset, which typically contains pictures depicting a non-rigid deformation and clutter (*e.g.*, in cat and person classes), is more significant. For example, the PCK gain over WS-SA [41] for the PF-PASCAL (0.79 vs. 0.72) is about two times more than that for the PF-WILLOW (0.74 vs. 0.71), indicating that our semantic flow method is more robust to non-rigid deformations and background clutter than semantic alignment approaches. (3) By comparing our model with a CNN-based semantic flow method [27], we can see that involving a spatial regularizer is significant. It focuses on designing fidelity terms (*e.g.*, using a contrastive loss [6]) only to learn a feature space preserving semantic similarities. This is because of a lack of differentiability of the flow field. In contrast, our model gives a differentiable flow field, allowing to exploit a spatial regularizer while further leveraging high-level semantics from CNN features more specific to semantic correspondence. (4) We confirm once more a finding in [31] that CNN features trained for ImageNet classification [9] clearly show the better ability to handle intra-class variations than hand-crafted ones such as SIFT [32] and HOG [7].

Caltech-101. The Caltech-101 [12] dataset, originally introduced for image classification, provides pictures of 101 image categories with ground-truth object masks. Unlike the PF [14, 15] and TSS [44] datasets, it does not provide ground-truth keypoint annotations. For fair comparison, we use 15 image pairs, provided by [16, 41], for each object category, and use the corresponding 1,515 image pairs for



Source image. Target image. CNNGeo [40]. A2Net [42]. WS-SA [41]. Ours.

Figure 3: Visual comparison of alignment results between source and target images on the PF-PASCAL dataset [15]. Keypoints of the source and target images are shown in diamonds and crosses, respectively, with a vector representing the matching error. All methods use the ResNet-101 features. Compared to the state of the art, our method is more robust local non-rigid deformations, scale changes between objects, and clutter. See text for details. (Best viewed in color.)

Type		Methods	LT-ACC	IoU
Hand-crafted	F	DeepFlow [39]	0.74	0.40
	F	GMK [10]	0.77	0.42
	F	SIFTFlow [30]	0.75	0.48
	F	DSP [26]	0.77	0.47
	F	HOG+PF-LOM [15]	0.78	0.50
	F	OADSC [46]	0.81	0.55
CNN-based	A	(T) VGG-16+A2Net [42]	0.80	0.57
	A	(T) ResNet-101+CNNGeo [40]	0.83	0.61
	A	(T+P) ResNet-101+WS-SA [41]	<u>0.85</u>	<u>0.63</u>
	F	(C+P) VGG-16+SCNet-AG+ [16]	0.79	0.51
	F	(B+P) FCSS+PF-LOM [27]	0.83	0.52
	F	(M) ResNet-101+Ours	**0.88**	**0.67**

Table 2: Quantitative comparison on the Caltech-101 dataset [12]. All numbers are taken from [15, 41, 42]. Numbers in bold indicate the best performance and underscored ones are the second best. C: Ground-truth correspondences.

evaluation. Following the experimental protocol in [26], we compute matching accuracy with two metrics using the ground-truth masks: Label transfer accuracy (LT-ACC) and the intersection-over-union (IoU) metric. Both metrics count the number of correctly labeled pixels between ground-truth and transformed masks using dense correspondences, where the LT-ACC evaluates the overall matching quality while the IoU metric focusses more on foreground objects. Following [41, 42], we exclude the LOC-ERR metric, since it measures the localization error of correspondences using object bounding boxes due to a lack of keypoint annotations, which does not cover rotations, affine or deformable transformations. The LT-ACC and IoU comparisons on the Caltech-101 dataset are shown in Table 2. Although this dataset provides ground-truth object masks, we do not retrain or fine-tune our model to evaluate its generalization ability on other datasets. From this table, we can see that (1) our model generalizes better than other CNN-based meth-

Mask consistency	Flow consistency	Smoothness	PCK ($\alpha = 0.1$)
✓	✗	✗	0.675
✗	✓	✗	0.718
✓	✓	✗	0.782
✓	✓	✓	**0.787**

Table 3: Average PCK comparison of different loss functions.

ods for other images outside the training dataset; and (2) it outperforms the state of the art in terms of the LT-ACC and IoU, verifying once more that our model focuses on regions containing objects while filtering out background clutter, even without using object proposals [15, 16, 27, 46] or an inlier counting [41].

Qualitative comparison. Figure 3 shows a visual comparison of alignment results between source and target images with the state of the art on the test split of the PF-PASCAL dataset [15]. We can see that our method is robust to a local non-rigid deformation (*e.g.*, bird's beaks and horse's legs in the first two rows), scale changes between objects (*e.g.*, front wheels in the third row), and clutter (*e.g.*, wheels in the last row). In particular, the fourth example clearly shows that our method gives more discriminative correspondences, cutting off matches for non-common objects. For example, it does not establish correspondences between a person and background regions in the source and target images, respectively, while others fail to cut off matches on these regions. We can also see that all methods do not handle occlusion (*e.g.*, a bicycle saddle in the last row).

4.4. Ablation study

We show an ablation analysis on different components and losses in our model. We measure PCK scores with height and width of the bounding box size, and report the results on the test split of PF-PASCAL dataset [15, 16, 41].

Training loss. We show the average PCK for three variants of our model in Table 3. The mask consistency term encourages establishing correspondences between prominent objects. Our model trained with this term only, however, may not yield spatially distinctive correspondences, resulting in the worst performance. A flow consistency term, which spreads flow fields over foreground regions, overcomes this problem, but it does not differentiate correspondences between background and objects. Accordingly, these two terms are complementary each other and exploiting both significantly boosts the performance of our model from 0.675/0.718 to 0.782, already outperforming the state of the art by a large margin (see Table 1). An additional smoothness term further boosts performance to 0.787.

Network architecture. Table 4 compares the performance of networks with different components in terms of the average PCK. The baseline models in the first three rows compute matching scores using both features from `conv4-23` and `conv5-3`, and estimate correspondences with different argmax operators. They do not involve any training similar to [31] that uses off-the-shelf CNN features for semantic correspondence. We can see that applying the soft argmax directly to the baseline model degrades performance severely, since it is highly susceptible to multi-modal distributions. The results in the next three rows are obtained with a single adaptation layer on top of `conv4-23`. This demonstrates that the adaptation layer extracts features more adequate for pixel-wise semantic correspondences, boosting performance of all baseline models significantly. Particularly, we can see that the kernel soft argmax outperforms others by a large margin, since it enables training our model end-to-end including adaptation layers at a sub-pixel level and is less susceptible to multi-modal distributions. The last three rows suggest that exploiting deeper level of features is important, and using all components with the kernel soft argmax performs best in terms of the average PCK.

Adaptation layer	Multi-level feature	Argmax Train	Argmax Test	PCK ($\alpha = 0.1$)
✗	✓	-	H	0.458
✗	✓	-	S	0.088
✗	✓	-	KS	0.284
✓	✗	S	H	0.725
✓	✗	S	S	0.717
✓	✗	KS	KS	0.750
✓	✓	S	H	0.768
✓	✓	S	S	0.762
✓	✓	KS	KS	**0.787**

Table 4: Average PCK comparison of different components. We denote by "H", "S", and "KS" hard, soft, and kernel soft argmax operators, respectively.

5. Conclusion

We have presented a CNN model for learning an object-aware semantic flow end-to-end, and introduced the corresponding CNN architecture, dubbed SFNet, with a novel kernel soft argmax layer that outputs differential matches at a sub-pixel level. We have proposed to use binary foreground masks directly to train a model for learning pixel-to-pixel correspondences that are widely available and can be obtained easily compared to pixel-level annotations. The ablation studies clearly demonstrate the effectiveness of each component and loss in our model. Finally, we have shown that the proposed method is robust to distracting details and focuses on establishing dense correspondences between prominent objects, outperforming the state of the art on standard benchmarks by a significant margin.

Acknowledgments. This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2017R1C1B2005584), the Louis Vuitton/ENS chair on artificial intelligence and the NYU/Inria collaboration agreement.

References

[1] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM TOG*, 28(3), 2009. 2

[2] Lubomir Bourdev and Jitendra Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *ICCV*, 2009. 2

[3] Hilton Bristow, Jack Valmadre, and Simon Lucey. Dense semantic correspondence where every pixel is a classifier. In *ICCV*, 2015. 1, 2

[4] Thomas Brox, Christoph Bregler, and Jitendra Malik. Large displacement optical flow. In *CVPR*, 2009. 1, 2

[5] Thomas Brox, Andrés Bruhn, Nils Papenberg, and Joachim Weickert. High accuracy optical flow estimation based on a theory for warping. In *ECCV*, 2004. 1, 2

[6] Christopher B Choy, JunYoung Gwak, Silvio Savarese, and Manmohan Chandraker. Universal correspondence network. In *NIPS*, 2016. 1, 2, 3, 5, 6

[7] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005. 1, 6

[8] Kevin Dale, Micah K Johnson, Kalyan Sunkavalli, Wojciech Matusik, and Hanspeter Pfister. Image restoration using online photo collections. In *ICCV*, 2009. 1

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009. 3, 5, 6

[10] Olivier Duchenne, Armand Joulin, and Jean Ponce. A graph-matching kernel for object categorization. In *ICCV*, 2011. 1, 6, 7

[11] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The Pascal Visual Object Classes (VOC) challenge. *IJCV*, 88(2), 2010. 5

[12] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE TPAMI*, 28(4), 2006. 5, 6, 7

[13] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *CVPR*, 2017. 5

[14] Bumsub Ham, Minsu Cho, Cordelia Schmid, and Jean Ponce. Proposal flow. In *CVPR*, 2016. 2, 3, 6

[15] Bumsub Ham, Minsu Cho, Cordelia Schmid, and Jean Ponce. Proposal flow: Semantic correspondences from object proposals. *IEEE TPAMI*, 40(7), 2018. 2, 5, 6, 7, 8

[16] Kai Han, Rafael S Rezende, Bumsub Ham, Kwan-Yee K Wong, Minsu Cho, Cordelia Schmid, and Jean Ponce. SCNet: Learning semantic correspondence. In *ICCV*, 2017. 1, 2, 3, 5, 6, 7, 8

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2, 3, 5

[18] Sina Honari, Pavlo Molchanov, Stephen Tyree, Pascal Vincent, Christopher Pal, and Jan Kautz. Improving landmark localization with semi-supervised learning. In *CVPR*, 2018. 4

[19] Asmaa Hosni, Christoph Rhemann, Michael Bleyer, Carsten Rother, and Margrit Gelautz. Fast cost-volume filtering for visual correspondence and beyond. *IEEE TPAMI*, 35(2), 2013. 1, 2

[20] Junhwa Hur, Hwasup Lim, Changsoo Park, and Sang Chul Ahn. Generalized deformable spatial pyramid: Geometry-preserving dense correspondence estimation. In *CVPR*, 2015. 1, 2, 5

[21] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015. 5

[22] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *NIPS*, 2015. 4

[23] Sangryul Jeon, Seungryong Kim, Dongbo Min, and Kwanghoon Sohn. PARN: Pyramidal affine regression networks for dense semantic correspondence estimation. In *ECCV*, 2018. 1, 2, 3, 4

[24] Angjoo Kanazawa, David W Jacobs, and Manmohan Chandraker. WarpNet: Weakly supervised matching for single-view reconstruction. In *CVPR*, 2016. 1, 2, 3, 5

[25] Alex Kendall, Hayk Martirosyan, Saumitro Dasgupta, Peter Henry, Ryan Kennedy, Abraham Bachrach, and Adam Bry. End-to-end learning of geometry and context for deep stereo regression. In *ICCV*, 2017. 4

[26] Jaechul Kim, Ce Liu, Fei Sha, and Kristen Grauman. Deformable spatial pyramid matching for fast dense correspondences. In *CVPR*, 2013. 1, 2, 5, 6, 7

[27] Seungryong Kim, Dongbo Min, Bumsub Ham, Sangryul Jeon, Stephen Lin, and Kwanghoon Sohn. FCSS: Fully convolutional self-similarity for dense semantic correspondence. In *CVPR*, 2017. 1, 2, 3, 5, 6, 7, 8

[28] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 6

[29] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, 2012. 2, 5

[30] Ce Liu, Jenny Yuen, and Antonio Torralba. SIFT flow: Dense correspondence across scenes and its applications. *IEEE TPAMI*, 33(5), 2011. 1, 2, 5, 6, 7

[31] Jonathan L Long, Ning Zhang, and Trevor Darrell. Do convnets learn correspondence? In *NIPS*, 2014. 2, 6, 8

[32] David G Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2), 2004. 1, 2, 6

[33] Simon Meister, Junhwa Hur, and Stefan Roth. UnFlow: Unsupervised learning of optical flow with a bidirectional census loss. In *AAAI*, 2018. 5

[34] David Novotny, Samuel Albanie, Diane Larlus, and Andrea Vedaldi. Self-supervised learning of geometrically stable features through probabilistic introspection. In *CVPR*, 2018. 5

[35] David Novotnỳ, Diane Larlus, and Andrea Vedaldi. AnchorNet: A weakly supervised network to learn geometry-sensitive features for semantic matching. In *CVPR*, 2017. 1, 2

[36] Masatoshi Okutomi and Takeo Kanade. A multiple-baseline stereo. *IEEE TPAMI*, 15(4), 1993. 1, 2

[37] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. 2017. 6

[38] Weichao Qiu, Xinggang Wang, Xiang Bai, Alan Yuille, and Zhuowen Tu. Scale-space SIFT Flow. In *WACV*, 2014. 2

[39] Jerome Revaud, Philippe Weinzaepfel, Zaid Harchaoui, and Cordelia Schmid. DeepMatching: Hierarchical deformable dense matching. *IJCV*, 120(3), 2016. 6, 7

[40] Ignacio Rocco, Relja Arandjelovic, and Josef Sivic. Convolutional neural network architecture for geometric matching. In *CVPR*, 2017. 1, 2, 3, 4, 5, 6, 7

[41] Ignacio Rocco, Relja Arandjelovic, and Josef Sivic. End-to-end weakly-supervised semantic alignment. In *CVPR*, 2018. 1, 2, 3, 4, 5, 6, 7, 8

[42] Paul Hongsuck Seo, Jongmin Lee, Deunsol Jung, Bohyung Han, and Minsu Cho. Attentive semantic alignment with offset-aware correlation kernels. In *ECCV*, 2018. 1, 2, 3, 4, 5, 6, 7

[43] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 2

[44] Tatsunori Taniai, Sudipta N Sinha, and Yoichi Sato. Joint recovery of dense correspondence and cosegmentation in two images. In *CVPR*, 2016. 1, 2, 5, 6

[45] Engin Tola, Vincent Lepetit, and Pascal Fua. DAISY: An efficient dense descriptor applied to wide-baseline stereo. *IEEE TPAMI*, 32(5), 2010. 1

[46] Fan Yang, Xin Li, Hong Cheng, Jianping Li, and Leiting Chen. Object-aware dense semantic correspondence. In *CVPR*, 2017. 2, 7, 8

[47] Hongsheng Yang, Wen-Yan Lin, and Jiangbo Lu. DAISY filter flow: A generalized discrete approach to dense correspondences. In *CVPR*, 2014. 1, 2

[48] Yi Yang and Deva Ramanan. Articulated human detection with flexible mixtures of parts. *IEEE TPAMI*, 35(12), 2013. 6

[49] Jure Zbontar and Yann LeCun. Computing the stereo matching cost with a convolutional neural network. In *CVPR*, 2015. 5

[50] Tinghui Zhou, Yong Jae Lee, Stella X Yu, and Alyosha A Efros. FlowWeb: Joint image set alignment by weaving consistent, pixel-wise correspondences. In *CVPR*, 2015. 1, 3

[51] Tinghui Zhou, Philipp Krahenbuhl, Mathieu Aubry, Qixing Huang, and Alexei A Efros. Learning dense correspondence via 3D-guided cycle consistency. In *CVPR*, 2016. 1, 2, 3, 5

[52] Yuliang Zou, Zelun Luo, and Jia-Bin Huang. DF-Net: Unsupervised joint learning of depth and flow using cross-task consistency. In *ECCV*, 2018. 5